

REMARKS

Claims 1-22 stand rejected under 35 U.S.C. § 103. Claims 1-2 and 12 have been amended.

Claims 1-9 and 11-22 stand rejected under 35 U.S.C. § 103 based on J. Akella et al.'s Synthesizing Converters Between Finite State Protocols (Akella) in view of U.S. Patent No. 5,680,522 issued to Netravali et al. (Netravali).

Akella discloses an interface created by a designer who manually enters the intended behavior of the interface in the form of a finite state machine. Akella does not disclose “automatically synthesizing an interface between the structurally different first and second protocols based on the first and second finite automata,” as recited in claim 1 as amended.

The systems described by Netravali have the same problems as Akella. For example, Netravali states that:

Conventional network interconnection systems typically have been built using ad hoc techniques. Problems in automatically synthesizing interconnections include overcoming network architectural mismatches and designing the protocol converters between the networks. Architectural mismatches involve differences in network layer functionalities, bit rates, buffering, interfaces and hand-off procedures. **Since considerable judgment, trade-off, analysis, and decision making is required to overcome architectural mismatches, automatic generation of systems to overcome the architectural mismatches is not foreseeable in the near future.**

(emphasis added)(Col. 1, lines 24-36). Examples of architectural mismatches, as given by Netravali, include “differences in network layer functionalities, bit rates, buffering, interfaces and hand-off procedures.” Clearly, because Netravali says that “automatic generation of systems to overcome the architectural mismatches (i.e. structural differences) is not foreseeable in the near future,” Netravali does not disclose nor suggest “automatically synthesizing an interface

between the structurally different first and second protocols based on the first and second finite automats, as recited in claim 1 as amended.

An example is on page 26 of the application, which states that “the exact correspondence between parts of the token is resolved, even if the data types of the two protocols are structurally different.” Another example of “automatically synthesizing an interface between the structurally different first and second protocols based on the first and second finite automats” can be found in the application as filed, page 6, lines 15-17, which states that the “present invention is also capable of generating the interface even when the data sequencing of the two protocols differs.” Another example can be found in the application as filed, page 11, which says that “the interface generator 118 determines 206 the permitted sequence of operations and resolves 208 all non-determinisms.” Another example is found on page 13, the “present invention does not require that the sequence in which the bits are transmitted be the same as the sequence that the bits be received.” Another example is on pages 27-28, as “a consequence, it may happen that the output of some piece of data must be delayed for many state transitions even though the data is already available in the registers of the interface.”

Even if Akella and Netravali were combined, the combination would neither teach nor suggest “automatically synthesizing an interface between the structurally different first and second protocols based on the first and second finite automats,” as recited in claim 1 as amended.

Therefore, applicants submit that claim 1 as amended is patentable over Akella in view of Netravali. Given that claims 2-11 and 21 depend from claim 1 as amended, applicants submit that these claims are also patentable over Akella in view of Netravali for at least this reason.

Claim 2 as amended recites “automatically corresponding data from said structurally different first and second protocols,” which is not disclosed by Akella and Netravali, either alone or in combination. Therefore, applicants submit that claim 2 as amended is patentable over Akella in view of Netravali for at least this reason.

Claim 3 recites “automatically translating data between said first protocol to said second protocol, said data in said first protocol having a first sequence, said data in said second protocol having a second sequence that is different from said first sequence,” which is not disclosed by Akella and Netravali, either alone or in combination. The examiner states that “Netravali discloses that it is to know to translate data between two protocols and rectify any mismatches.” Netravali does not disclose this. In fact, Netravali discloses the opposite, “automatic generation of systems to overcome the architectural mismatches (i.e. different sequences) is not foreseeable in the near future.” Therefore, applicants submit that claim 3 is patentable over Akella in view of Netravali for at least this reason.

Claim 10 stands rejected under 35 U.S.C. § 103 based on Akella in view of Netravali and in further view of Mano's Computer System Architecture (Mano). Akella, Netravali, and Mano, either alone or in combination, neither disclose nor suggest “identifying non-deterministic transitions for each interface state; selecting a single outgoing transition for each interface state for each input value based upon priority parameters to generate a deterministic interface between the first and second protocols.” as recited in claim 10.

In the rejection of claim 10, the examiner states that “Claims 1 and 8’s arguments apply... identifying the non-deterministic transition and determine a transition state for each input are fundamental practices in FSM’s construction.” Claim 1’s argument says “The Akella’s computer design creates a product machine, which is pruned of invalid/useless states. Since the

third FSM only represents the valid data operations between the two protocols, and each protocol's invalid operations or unconvertible operations become non-deterministic (invalid/useless), it is clear that the third FSM only processes the operations, which are deterministic, and thus discards the un-deterministic operations.

Applicants submit that the invalid/useless operations of Akella are not non-deterministic transitions for each interface state as recited in claim 10. The non-deterministic transitions for the interface state mean that the state has more than one transition. In fact, the invalid/useless operations of Akella are deterministic, because there is no question that they cannot be used. Therefore, Akella does not disclose "identifying non-deterministic transitions for each interface state," as recited in claim 10. Therefore, applicants submit that claim 10 as amended is patentable over Akella in view of Netravali and Mano.

Claim 12 stands rejected based on Akella in view of Netravali. Akella and Netravali, either alone or in combination, neither disclose nor suggest "a synthesizing unit to automatically synthesize an interface between the structurally different first and second protocols based on the first and second finite automata," as recited in claim 12 as amended. Therefore, applicants submit that claim 12 as amended is patentable over Akella in view of Netravali. Given that claims 13-19 depend from claim 12 as amended, applicants submit that these claims are also patentable over Akella in view of Netravali.

Claim 20 stands rejected based on Akella in view of Netravali. The Office action states that:

The Akella's computer design creates a product machine, which is pruned of invalid/useless states (specification, page 5, line 5). Since the third FSM only represents the valid data operations between the two protocols, and each protocol's invalid operations or un-convertible operations become non-deterministic,

(invalid/useless), it is clear that the third FSM only processes the operations, which are deterministic, and thus discards the non-deterministic operations.

However, applicants submit that the “product machine is taken and pruned of the invalid/useless states” as disclosed on page 5 of the application does not disclose “automatically eliminating non-determinisms” as recited in claim 20. An invalid state is not a non-deterministic state. A non-deterministic state occurs when two or more operations may be used. A deterministic state occurs when only one operation can be used. In fact, an invalid/useless state is deterministic, because it is not used. Therefore, Akella does not disclose “automatically eliminating non-determinisms in said third representation” as recited in claim 20.

Even if Akella and Netravali were combined, the combination would neither disclose nor suggest “automatically eliminating non-determinisms in said third representation” as recited in claim 20. Therefore, applicants submit that claim 20 is patentable over Akella in view of Netravali.

Claim 21 recites “automatically generating a third representation, representing one or more permitted operations of said first and second finite automata.” In both Akella and Netravali, the permitted operations have to be specified by a user. Therefore, applicants submit that claim 21 is patentable over Akella in view of Netravali for at least this reason.

Claim 22 stands rejected based on Akella in view of Netravali. Akella and Netravali, alone or in combination, neither disclose nor suggest “automatically eliminating at least one non-determinism in the representation” as recited in claim 22. Therefore, applicants submit that claim 22 is patentable over Akella in view of Netravali.

CONCLUSION

Allowance of the claims is respectfully requested. The Examiner may call the Assignee's attorney at (650) 849-4422 to further advance prosecution of this case to issuance.

If the Commissioner determines that additional fees are due or that an excess fee has been paid, the Patent Office is authorized to debit or credit (respectively) Deposit Account No.

50-2518, billing reference no. 7010602001.

DATED: April 14, 2004

Respectfully submitted,

By: Jeffrey S. Smith
Jeffrey S. Smith
Registration No. 39,377

Bingham McCutchen LLP
Three Embarcadero Center, Suite 1800
San Francisco, California 94111
Telephone: (650) 849-4422
Telefax: (650) 849-4800